CLAIMS

What is claimed is:

1. An IPv6 header receiving apparatus comprising:

a register with a data size that is a multiple of an octet, which receives IPv6 header data in units of an octet, stores the IPv6 header data, and transmits the stored IPv6 header data to an Ipv6 processing module that corresponds to the IPv6 header data; and

modules for an IPv6 basic header or various types of IPv6 extended headers, which receive the IPv6 header data from the register and process the IPv6 header data.

2. An IPv6 header receiving apparatus comprising:

a counter which counts up to a specified amount of IPv6 extended header data that is transmittable in units of an octet at a time;

a register with a data size that is a multiple of an octet, which receives IPv6 header data in units of an octet, stores the IPv6 header data, and, if the counter has completed counting up to the specified amount, transmits the stored IPv6 header data to a module for processing an IPv6 header that corresponds to the IPv6 header data; and

modules for an IPv6 basic header or various types of IPv6 extended headers, which receive the IPv6 header data from the register and process the IPv6 header data.

3. An IPv6 header receiving apparatus comprising:

an octet indicator which counts up to an amount of IPv6 extended header data that can be transmitted in units of an octet at a time;

a register with a data size that is a multiple of an octet, which receives IPv6 header data in units of 8 octets and stores the IPv6 header data;

a control unit, which analyzes the IPv6 header data stored in the register to determine a type and length corresponding to the IPv6 header data, and, if the octet indicator has completed counting up to the specified amount, instructs the register to transmit the IPv6 header data with the determined length to an IP header processing module that corresponds to the IPv6 header type; and

modules for an IPv6 basic header or various types of IPv6 extended headers, which receive the IPv6 header data from the register and process the IPv6 header data.

4. The IPv6 header receiving apparatus of claim 3, wherein the register comprises:

a temporary register with a data size that is a multiple of an octet, which receives IPv6 header data in units of an octet and stores the IPv6 header data; and

a shift register, which receives the IPv6 header data in units of an octet each time from the temporary register and, if the shift register is filled with an amount of data to be transmitted at one time to a module, the shift register transmits the filled amount of data to the modules.

5. The IPv6 header receiving apparatus of claim 4, wherein the control unit comprises:

a header analyzer which receives data from the temporary register and analyzes the data;

a next header status register which stores next header type information following a current header, based on the information obtained from the analysis by the header analyzer;

a length register which stores header length information of the next header, based on the information obtained from the analysis; and

a path determiner which instructs the shift register of the amount of data to be transmitted and of the module to receive the data, the data amount specified in the header length information stored in the length register and the module specified in the next header type information stored in the next header status register.

6. The IPv6 header receiving apparatus of claim 5, wherein the octet indicator comprises:

an auxiliary counter which counts up to a data amount stored in the shift register; and

a main counter which counts up to a data amount corresponding to the header length information stored in the length register, and the control unit instructs the temporary register to receive a next packet of data if a current count of the main counter exceeds a value corresponding to the header length information.

7. The IPv6 header receiving apparatus of claim 3, wherein the modules are a basic header module, a routing header module, a destination option header module, an authentication header module, an ESP header module, a hop-by-hop header module, and an upper layer module.

8. An IPv6 header processing method comprising:

filling a register with IPv6 header data received in units of a predetermined size, which is a multiple of an octet;

identifying an IPv6 header type by analyzing the IPv6 header data filled in the register; and

transmitting the IPv6 header data to a module corresponding to the identified IPv6 header type.

9. An IPv6 header processor comprising:

a data link layer, which transmits data transmissions;

an IPv6 controller, which is responsive to header data of the data transmissions from the data link layer, and detects a type and length of the header data, and outputs the header data based on the type and length of the header data detected in the data transmissions; and

a register file having a plurality of IPv6 header modules, coupled to the IPv6 controller, each IPv6 header module receives and processes the corresponding header data transmitted by the IPv6 controller.

10. The processor of claim 9, wherein the IPv6 controller comprises:

storage registers to store the header data of data transmissions;

a counter which increments a value of an indicator when an octet of the header data is received; and

a control unit which detects the type and length of the header data, and outputs the header data from the storage registers based on the indicator value, to the corresponding IPv6 header module, wherein the corresponding IPv6 header module is determined based on the detected type and length of the header data, wherein the corresponding IPv6 header module processes the header data.

11. The processor of claim 10, wherein the storage registers and the plurality of header processors are each multiples of an octet.

12. The processor of claim 10, wherein the storage registers comprise:

a buffer register to receive data transmissions in multiples of an octet; and

a transmit register, wherein the buffer register outputs octets of header data to the transmit register where the header data is stored, and when the indicator value is equivalent to a predetermined value the contents of the transmit register are output under direction of the control unit.

13. The processor of claim 12, the control unit comprises:

a header analyzer, which receives data from the buffer register and determines a type and length of the header data;

a next header status register which stores the type of the header data determined by the header analyzer;

a length register which stores the length the header data determined by the header analyzer; and

an output path determiner responsive to the contents of the next header status register and the length register, which directs the contents of the transmit register to the corresponding header processor.

14. A method of processing header data comprising:

shifting header data into a first register in packets of lengths that are multiples of an octet;

transmitting the header data into a second register where the header data is maintained;

determining a type and length of the header data, which determines the output path of the header data maintained in the second register;

incrementing a counter each time the header data is transmitted to the second register from the first register; and

shifting the contents of the second register to a predetermined processing module by the determined output path when the counter reaches a predetermined value.

15. The method of claim 14, wherein the header data is in IPv6 format.

16. The method of claim 15, further comprising:

counting a maximum effective length of each header data; and

determining whether the maximum effective length of each header data exceeds a predetermined value, wherein if the predetermined value is exceeded a next header data packet is received.

17. The method of claim 16, wherein if the predetermined value is not exceeded, additional header data is shifted into the first register.

18. The method of claim 16, further comprising:
receiving the header data from a media access control (MAC) layer.